B2S: A Program to Reconstruct Geometrical Information from Computer Aided Design Models (CAD) and Converting into Primitive Mathematical Form

P.V. Subhash[1], C.V. Suresh [2,3], Shrichand Jakhar[3], C.V.S. Rao[3], T.K. Basu[4]

[1] ITER-India, Institute for Plasma Research, Gandhinagar, India

[2] University College of Engineering, Vizianagaram , J.N.T.U. Kakinada, AP, India

[3] Institute for Plasma Research, Gandhinagar, India

[4] Raja Ramanna Fellow, DAE, India

[1] getsubhash@gmail.com

Abstract

Presently many of the experimental systems are designed using Computer Aided Design (CAD) software. But the physics and mathematical modelling with traditional computational codes require geometrical information in conventional manner. Extracting the geometrical information from a complicated CAD model is non-trivial, tedious and time consuming process. A computer code has been developed to reconstruct the geometrical information from Computer Aided Design (CAD) models. CAD models generally occupy the geometric class called Boundary representation (Brep), thus the present code converts boundary representations (Brep) into Constructive Solid Geometries (CSG). The present code extracts geometric information from CAD models and converts it to primitive mathematical form which can then be readily used in other codes like fluid codes, radiation transport codes. This code can also be used for engineering applications like geometric recognition. The present code acts as a bridge between the modern CAD representation and primitive geometrical representation required by physicists or mathematicians. The code can be made available freely for academic use upon request.

Keywords

CAD; Tokamks; Neutronics

Introduction

Advances in Computer Aided Design (CAD) have greatly changed the design methodology of describing geometries of experimental systems, reactors, automobiles etc. Presently most of the complicated experimental set-ups are initially designed using CAD software like Catia, Solidworks etc. It is easy to demonstrate and understand these models in CAD software. But proper geometric understanding and mathematical interpretation of the models require expertise in CAD software. Further regenerating geometrical information for complicated models which contain few hundreds of geometries demand lot of time and effort. In many cases computational modelling of the experiments is done by physicists who are not necessarily familiar with CAD software. Most physical modelling is carried out using many numerical algorithms, which is not directly compatible with CAD models. For example many of the fluid dynamics codes, radiation transport codes and many other codes require geometries in mathematical form rather than those in CAD form. In some cases the analyser may only be interested in one among many geometries. For example, a bolt manufactures might be only interested in geometric information of the corresponding 'grooves' among the many geometries presented in the whole model.

Thus an algorithm for conversion of CAD to geometric model in mathematical form is warranted and useful for general scientific community as well as in engineering production fields. In the production field, geometrical extraction and detection of specific geometries are few among the many applications. As an example of the application in physics, we can consider the following example. Radiation transport codes like Monte-Carlo N-Particle (MCNP) require an input file in primitive mathematical form. In many

cases the radiation modelling is performed after designing the experimental system using CAD software. The modelling is in general carried out by physicists who may not be familiar with CAD software. Even for experts in CAD software, it will be a time consuming task to convert the CAD geometries into primitive mathematical forms. Further any modification in basic CAD model leads to repetition of the same task again.

The process of reconstruction of geometric information from CAD models has been reported for some specific purposes. Reference reports such an effort for machining planar surfaces. Some of the studies for specific pattern recognition have been reported in References. H.Lau et al. reported an algorithm for conversion of CAD to Computer Aided Process Planning (CAPP) (see References). A method for separation of boundary of CAD geometry has been reported in Reference and a conversion methodology form CSG to two dimensional solid is presented in References. Most of the reported studies are limited to some specific applications and not much attampt is made to obtain the primitive mathematical forms of the original model. The advantage of obtaining the geometries in primitive mathematical form makes many physics computer codes more 'user friendly'.

One may adopt one of the two methodologies for conversion of CSG Brep to CSG. In the first method, the entire model will be divided into large number of small blocks. Boolean operators are then used for the conversion (see References). Secondly one can directly interpret the solid models from the neutral format and the information can be used for the mathematical representation of the model. This paper reports a program to convert the CAD geometries into the primitive mathematical forms using the second method described above.

Details of Algorithm

In general, CAD software produces output files that can be saved in many formats like IGES, prt, STEP etc. Among these, STEP is a universally accepted neutral format. It is possible to convert any format into STEP format and vice-verse. The information is represented in the form of grammar free English, which can be interpreted. The geometric information is encoded in a specific format which has to be decoded systematically. The STEP file stores the information in three parts: descriptive, geometric and topological.

An algorithm is developed for to extract the geometrical information from a STEP file. A detailed explanation is given below. The processes of converting the CAD geometries into primitive mathematical forms have been done in three steps. First step is the extraction of geometric data from the STEP file. In the second step the extracted data is mathematically processed to obtain the geometric data and topological information into the fully linked geometric information. Finally the reconstructed geometric data is converted into a mathematical form which is easy to understand by the users. The data flow structure that the present algorithm follows is explained below.

In STEP file the entire geometrical and topological data is stored in hierarchical tree which follows the 'bottom-up' approach. The very starting root element/node is the CLOSED SHELL and the last terminating element/node is the CARTESIAN POINT. In between these nodes, there are six more nodes which represent the entire data flow structure in the STEP file. The required geometrical data can be extracted from the top element to the bottom element with the main root functional element as CLOSED SHELL at the top level and the main terminating functional element as CARTESIAN POINT at the bottom level. To make connectivity among these two levels, the entire instance addresses are processed at each level.

The CLOSED SHELL (CS) defines the outer extent of the solid geometry. This is a type of connected face set, called ADVANCED FACEs. The CLOSED SHELL instance stores the corresponding instances of the ADVANCED FACEs (AF). The CS should have minimum of one AF instance information. Each AF contains two types of data, the FACE OUTER BOUND (FOB) information and some time the nature of the surfaces like PLANE, SPHERICAL, CYLINDRICAL, CONICAL, TOROIDAL, BOUNDED surface etc. Each AF must have at least one FOB instance information and one surface instance information which is unique.

FOB is a loop used to bound one ADVANCED FACE and this provides two instances, in which the first gives information of the higher level called EDGE LOOP (EL) while the second in AF gives the information about the type of surface. Further the second instance is traced to extract some of the geometrical data for the corresponding AF.

EDGE LOOP is a loop that bounds the face by a collection of edges. It is a closed path in which the start and end vertices are the same to close the loop around the face. The loop formed by a group of straight line segments is called polyglot, in which straight lines bound a planar region. Higher order edges are represented either in interpolated form or through higher order loops. The EL instance consists of a list of ORIENTED EDGES (OE) which are constructed from another edge so as to form a closed bound, around the face. Also it gives the orientation of the AF. Each EL is further processed to obtain the instances of the OE.

Every OE of the corresponding EL is used to find out the EDGE CURVE (EC) instance included in each OE. The EDGE CURVE is one edge which has fully defined geometry with the VERTEX POINTs (VP). This EC also gives the information about the planar orientation (direction) of the EC. Each EC instance will be processed and the corresponding VP instances are stored.

VERTEX POINT is a point defining the geometry of a vertex. Each VP is then traced to store the corresponding CARTESIAN POINT (CP) instances.

CARTESIAN POINT is the address of a point in the space in the given coordinate system. This CPs are stored in an array for further processing. Once the entire CP for every geometry is completely extracted then the data are passed for mathematical processing.

Once the primary geometric data and topological information are obtained from the model, mathematical processing is warranted for proper reconstruction of geometric models. The geometric information is stored inside the STEP file either in direct mathematical forms or in interpolated forms. If it is in the form of interpolated data, it will be separately processed to obtain nature of the surface which bounds the geometries and other geometric information like co-ordinates.

Finally, the re-constructed geometries will be stored as output in general mathematical form. The essential information in the output is nature of the geometries, number of surfaces bounding the geometric and mathematical forms of geometry and surfaces. Here all the surfaces are given in general mathematical forms. For example, for a planar surface all the four Cartesian points as well as the plane equation will be given. All the higher order surfaces in the output file will be in the form of a general quadratic equation as given below

$$Ax^2+By^2+Cz^2+Dxy+Eyz+Fxz+Gx+Hy+Jz+K = 0 \quad (1)$$

All the above 10 co-efficient (A to K) along with other relevant geometric information will be stored in the output file.

Detailed explanation of the input and output files and other details are explained in the following Section

Input-output Format

In this section we will explain the structure of input and output files in detail. The input for the code should be supplied through the input file called 'inputs.in'. The format of the input file is very simple and self-explanatory in nature. The input step file should be kept in a directory named INPUT (The user can also place it in root directory, in that case proper path should be provided in inputs.in file). The file name should be entered in the input file. The second entry in the input file is about the nature of the conversion. The code can be used to convert the entire geometry into mathematical form or only selected type of geometry.

In the present form the code can handle following geometries.

- Planar Solids
- Spheres
- Circular Cylinders
- Circular Cones
- Ellipsoids
- Elliptical cylinders
- Torus

The mathematical information about each kind of surfaces is stored in separate output files. For example output file name for all the geometries bounded by spherical surface will be stored in the output file called 'sphere.txt'. Mainly the output file gives two kinds of data. Firstly the geometric data include nature of the geometries and other important geometric information like coordinates of important locations in the geometries. Secondly a mathematical equation of the present geometry will be presented in the output file. Normally the information in mathematical forms is expressed either as first degree equations (Planes) of the form

$$AX + BY + CZ - D = 0 \quad (2)$$

or general second degree equations in the form of equation 1. Detailed explanations of the output format for all type of geometries are given below.

Planar Solids

The corresponding output file name is 'plane.txt' and this contains information regarding the geometries bounded by planar surfaces. Mainly two kind of geometric inflammations are provided, firstly the co-ordinates of all corners and secondly the mathematical equation of the corresponding plane as in equation 2.

Spheres

The following information is supplied in the output file 'sphere.txt'.

– Co-ordinate of the sphere centre.

– Radius of the sphere.

– Co-efficient of the general equation 1 of the corresponding sphere.

Circular Cylinders

A circular cylinder contains three surfaces, one cylindrical surface and two circular planar surfaces. The output file 'cylinder.txt' provides required information about these three surfaces. The important information extracted from the output file is summarized below:

- Co-ordinates of the cylinder centre.
- Radius and length of the cylinder
- Co-ordinates of the top and bottom circular planes centre.
- Orientation of the axis of the cylinder
- Generalized equation of the cylinder as in the equation 1
- Plane equations for two planes as in the equation 2

Circular Cone

The output file corresponding to circular cone is named as 'cones.txt'. The information about conical surface as well as the circular surface that bounds the cone is provided in this output file. A summary of the geometric information in the file 'cones.txt' is given below:

- The vertex point of the cone.
- Radius and length of the conical surface.
- Co-ordinates of the bounding circular planes centre.
- Orientation of the axis of the cone
- Generalized equation of the cone as in the equation 1
- Plane equations for plane as the equation 2

Ellipsoids

Following important information will be provided through the output file name 'ellipsoid.txt'

- Axis lengths along x, y, z directions.
- Centre point of the ellipsoid
- Orientation of the axis.
- Generalized equation of the ellipsoid as in the equation 1

Elliptical Cylinders

The output file called 'ellipticalcylinders.txt' provides the following required mathematical information of elliptical cylinders:

- Axis lengths along x, y, z directions and length of the cylinder
- Centre point of the elliptical planes.
- Orientation of the axis
- Generalized equation of the elliptical cylinder as in the equation 1
- Plane equations for two elliptical plane as in the equation 2

Torus

Both the major and minor radius as well as the centre point of the torus will be provided through the output file called 'torus.txt'. It should be noted that for the particular case of torus no information regarding the orientations is available at present. This is a restriction of the present code.

Running the Code

Firstly the CAD model (parts, assemblies etc.) should be saved as part file using any CAD software. Then the part file should be re saved in to STEP file format. This STEP file can be directly imported into the present code through the input file inputs.in. After compilation of the file b2s.f, user can run the executable 'a.out' to get the output. Even through this code is made to use in Linux/UNIX operating systems,

with some small additional pre-processing, it can be used in Windows operating system as well. Two changes should be done as follows. Firstly the Linux specific lines in source program (which has been separately notified) should be commented. Secondly the user should replace all '#' in STEP file with '#' (A space after #).

Test Runs

The example which demonstrates the working of the code has been done against two dummy systems created in CAD software. The code has been tested both in Linux and windows environment using f77 and g77 FORTRAN compiler respectively.

Problem 1

The 2D representation of CAD file is shown in FIG. 1. The figure shows a dummy in house experimental system. It should be noted that the system shown in FIG. 1 is not a real experimental system, but created only for the demonstration purpose. The system consists of a glass window house and an imaginary apparatus in cylindrical and ellipsoidal shapes. Few decorative elements made from cones and spheres are also present in the model.

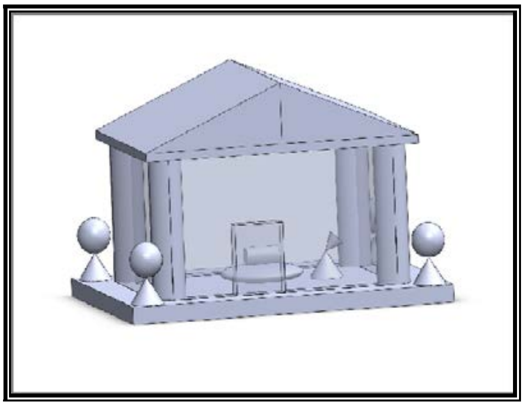

FIG. 1 GEOMETRIC MODEL USED IN PROBLEM-1

It can be noted that we used the option 'A' in the option column, which means that all geometries will be converted. The code will create following five output files.

- cones.txt
- cylinder.txt
- ellipsoid.txt
- planes.txt
- sphere.txt

All the output files are presented in the appendix

Problem 2

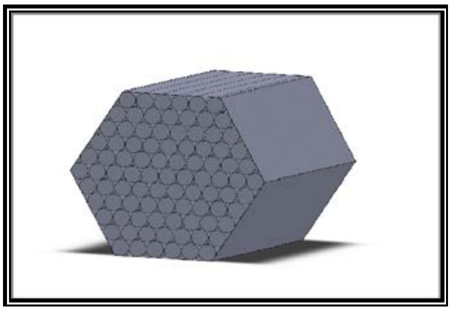

FIG. 2 GEOMETRIC MODEL USED IN PROBLEM-2

FIG. 2 shows the geometry treated in this example. Here we showed a hexagonal container containing 91 cylindrical roads. We now use the option 'P' in the input file, which contains only planar surfaces. This is an example of geometrical recognition problem. The corresponding input and output files are also shown in appendix

One set of work with this code can be seen from Reference. The code may be made available for academic uses upon request. The code is tested with CAD outputs from SolidWorks software, but as STEP format is a universal format the code is expected to work across other CAD software also.

Conclusions

A code to convert CAD models into primitive mathematical form has been developed. The code takes CAD models in STEP format as input and produces an output which contain details of the geometries and their mathematical forms. The present code extracts the geometric data from the CAD model and process mathematically to reconstruct the geometric information. Which is then given in primitive mathematical form through the output file. The code will be particularly useful for scientists and engineers to get the geometric information from complicated CAD models for various types of codes like fluid dynamics code, radiation transport codes etc. The other applications are geometry recognition for process planning and other similar engineering applications.

Further work is under way to extend the code to produce outputs in a format compatible with Monte-Carlo-N-Particle (MCNP) code.

ACKOWLEDGEMENT

C.V. Suresh acknowledges Institute for Plasma Research for the internship provided by the National

Fusion Program fellowship no:IPR:EST:NFP- INT-S:7:JAN-2010.

REFERENCES

G.Tsuzuki, M. de Sales, Takase, F.K., S.Garcia, M.A., Martins, T. de Castro, Journal of the Brazilian Society of Mechanical Sciences and Engineering4, 337 (2007).

Ismail, N., Osman, M., Tan, C.,S., Wong, Sulaiman, S., Internat onal Journal of Engineering and Technology1, 206 (2004).

Kumar, A., Saha, J., ARISER-Industrial Engineering4, 129 (2008).

Lau, H., Jiang, B., Computer Integrated Manufacturing Systems11, 67 (1998).

Liang, M., Ahamed, S., den Berg, B., Computers in Industry32, 219 (1996).

Shapiro, V., Vossler, D.L., ACM Transactions on Graphics12, 35 (1993).

Shapiro, V., Vossler, D.L., Journal of Mechanical Design113, 292 (1991).

Subhash, P.V., Thomas, Sajal, Jakhar, S., Suresh, C.V., Sati, U., Vettur, Ramesh, Rao, C.V.S. , Basu, T.K. and Vasu, P., 23rd ITPA topical meeting on diagnostics, November 27-30, Gandhinagar, Gujarat., (2012).

Tan, C.F., Tan, N.x, Wong, S.V., Sulaiman, S., Osman, M.R., Jurnal Teknologi43, 73 (2007).

P.V. Subhash was born in India on 1980. Dr. Subhash obtained bachelor and master degrees in Physics from University of Calicut, India in the years 2000 and 2003 respectively. Dr. Subhash acquired PhD degree in Plasma physics from Institute for Plasma Research, Gandhinagar, India in 2011.

Dr. Subhash is working as a Scientist at ITER-India, Institute for Plasma Research, India since 2009.

Dr. Subhash is a life member of Plasma Science Society of India (PSSI) and a member of Indian Science Congress Association.

APPENDIX

Problem-1

```
$inputs
input = 'INPUT/TEST1.STEP'
option = 'A'
$
C option = 'A' for ALL Surfaces
C option = 'P' for Plane Surfaces
C option = 'S' for Spherical Surfaces
C option = 'C' for Cylindrical Surfaces
C option = 'CO' for Conical Surfaces
C option = 'E' for Ellipsoid Surfaces
C option = 'EC' for Elliptical Cylinder Surfaces
C option = 'T' for Circular Torus
```

Output files

cones.txt}

```
######################################################################
THIS IS THE OUTPUT FILE FOR CONICAL SURFACES.
NUMBER OF GEOMETRIES BOUNDED BY ATLEAST ONE CONICAL SURFACE &
ONE CIRCULAR PLANE IS A CONE.
OUTPUT FORMAT OF ONE CONE:
  1. CONE CIRCLE RADIUS (r) WILL BE PROVIDED.
  2. CONE LENGTH (l) WILL BE PROVIDED.
  3. VERTEX POINT OF CONE (x1, y1, z1) WILL BE PROVIDED.
  4. CENTER POINT OF CONE CIRCLE (x2, y2, z2) WILL BE PROVIDED.
  5. ANGLES WITH RESPECT TO X-,Y-,Z- AXIS WILL BE PROVIDED.
  6. COEFFICIENTS OF GENERALIZED QUADRATIC EQUATION IN THE FORM OF
     AX^2 + BY^2 + CZ^2 + DXY + EYZ + FXZ + GX + HY + JZ + K = 0
     A, B, C, D, E, F, G, H, J, K WILL BE PROVIDED.
  7. COEFFICIENTS OF PLANE EQUATION IN THE FORM OF:: PX+QY+RZ-S=0
     (P, Q, R, S)WILL BE PROVIDED.
######################################################################
```

CONE NUMBER : 1

CONE RADIUS (r) :: 10. & CONE LENGTH (l):: 15.

VERTEX & CENTER POINTS

(X1, y1, z1) :: 110. -65. 60.

(x2, y2, z2) :: 110. -80. 60.

ANGLES WITH RESPECT TO

X - AXIS :: 90.0000011

Y - AXIS :: 180.000002

Z - AXIS :: 90.0000011

COEFFICIENTS (A,B,C,D,E,F,G,H,J,K)

-125. 100. -125. 0. 0.

0. 27500. 13000. 15000. -1540000.

COEFFICIENTS OF PLANE EQUATION

------------------------------(P,Q,R,S) :: 0. 100. 0. -8000.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

CONE NUMBER : 2

CONE RADIUS (r) :: 5. & CONE LENGTH(l) :: 10.

VERTEX & CENTER POINTS

(x1, y1, z1) :: 110. -65. 60.

(x2, y2, z2) :: 117.071068 -57.92 60.

ANGLES WITH RESPECT TO

X - AXIS :: 45.0000005

Y - AXIS :: 45.0000005

Z - AXIS :: 90.0000011

COEFFICIENTS (A,B,C,D,E,F,G,H,J,K)

-25. -25. -75. 100. 0.

0. 12000. -14250. 9000. -1393125.

COEFFICIENTS OF PLANE EQUATION

(P, Q, R, S):: -17.65 -17.7 0. -1040.92 >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

CONE NUMBER : 3

CONE RADIUS (r) :: 10. & CONE LENGTH (l) :: 15.

VERTEX & CENTER POINTS

(x1, y1, z1) :: -18. -65. 128.

(x2, y2, z2) :: -18. -80. 128

ANGLES WITH RESPECT TO

X - AXIS :: 90.0000011

Y - AXIS :: 180.000002

Z - AXIS :: 90.0000011 COEFFICIENTS (A,B,C,D,E,F,G,H,J,K)

-125. 100. -125. 0. 0.

0. -4500. 13000. 32000. -1666000

COEFFICIENTS OF PLANE EQUATION

(P, Q, R, S) :: 0. 100. 0. -8000.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<< CONE NUMBER : 4

CONE RADIUS (r) :: 10. & CONE LENGTH (l) :: 15.

VERTEX & CENTER POINTS

(x1 ,y1, z1) :: 158. -65. 128.

(x2, y2, z2) :: 158. -80. 128.

ANGLES WITH RESPECT TO

X - AXIS :: 90.0000011

Y - AXIS :: 180.000002

Z - AXIS :: 90.0000011

COEFFICIENTS (A,B,C,D,E,F,G,H,J,K)

-125. 100. -125. 0. 0.

0. 39500. 13000. 32000. -4746000.

COEFFICIENTS OF PLANE EQUATION

(P, Q, R, S) :: 0. 100. 0. -8000.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

CONE NUMBER : 5

CONE RADIUS (r) :: 10. & CONE LENGTH(l) :: 15.

VERTEX & CENTER POINTS

(x1, y1, z1) :: -18. -65. -7.

(x2, y2, z2) :: -18. -80. -7.

ANGLES WITH RESPECT TO

X - AXIS :: 90.0000011

Y - AXIS :: 180.000002

Z - AXIS :: 90.0000011

COEFFICIENTS (A,B,C,D,E,F,G,H,J,K)

------------------------------------ -125. 100. -125. 0. 2.664536E-014

0. -4500. 13000. -1750. 375875.

COEFFICIENTS OF PLANE EQUATION

(P, Q, R, S) :: 0. 100. 0. -8000.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<< CONE NUMBER : 6

CONE RADIUS (r) :: 10. & CONE LENGTH (l) :: 15.

VERTEX & CENTER POINTS

(x1, y1, z1) :: 158. -65. -7.

(x2, y2, z2) :: 158. -80. -7.

ANGLES WITH RESPECT TO

X - AXIS :: 90.0000011

Y - AXIS :: 180.000002

Z - AXIS :: 90.0000011

COEFFICIENTS (A,B,C,D,E,F,G,H,J,K)

-125. 100. -125. 0. 2.66456E-014 0. 39500. 13000. -1750. -2704125.

COEFFICIENTS OF PLANE EQUATION

(P, Q, R, S) :: 0. 100. 0. -8000.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

Cylinder.txt

##

THIS IS THE OUTPUT FILE FOR CIRCULAR CYLINDRICAL SURFACES.

NUMBER OF GEOMETRIES BOUNDED BY ATLEAST ONE CYLINDRICAL SURFACE &

TWO CIRCULAR PLANES (TOP & BOTTOM) IS A CYLINDER.

OUTPUT FORMAT OF ONE CYLINDER:

1. CYLINDER RADIUS (r) WILL BE PROVIDED.

2. CYLINDER LENGTH (l) WILL BE PROVIDED.

3. ANGLES WITH RESPECT TO X-,Y-,Z- AXIS WILL BE PROVIDED.

4. CENTER POINT OF TOP & BOTTOM CIRCLES (x1, y1, z1) & (x2, y2, z2)

WILL BE PROVIDED.

5. COEFFICIENTS OF GENERALIZED QUADRATIC EQUATION IN THE FORM OF

AX^2 + BY^2 + CZ^2 + DXY + EYZ + FXZ + GX + HY + JZ + K = 0

A, B, C, D, E, F, G, H, J, K WILL BE PROVIDED.

6. COEFFICIENTS OF PLANE EQUATIONS AT TOP, BOTTOM

IN THE FORM OF :: PX+QY+RZ-S=0,

(P1, Q1, R1, S1) & (P2, Q2, R2, S2) WILL BE PROVIDED.

##
#####################

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

CYLINDER NUMBER : 1 CYLINDER RADIUS (r) :: 10. & CYLINDER LENGTH (l) :: 80

ANGLES WITH RESPECT TO

X - AXIS :: 90.0000011

Y - AXIS :: 180.000002

Z - AXIS :: 90.0000011

CENTER POINTS

----------------------- (x1, y1, z1) :: 140. 0. 110. (x2, y2, z2) :: 140. -80. 110.

COEFFICIENTS (A,B,C,D,E,F,G,H,J,K)

6400. 0. 6400. 0. 0.

0. -1792000. 0. -1408000. 20224e4 COEFFICIENTS OF TOP & BOTTOM PLANE EQUATIONS

(P1, Q1, R1, S1) :: 0. -100. 0. 0.

(P2, Q2, R2, S2) :: 0. -100. 0. 8e3.

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

CYLINDER NUMBER : 2

CYLINDER RADIUS (r) :: 10. & CYLINDER LENGTH (l) :: 80.

ANGLES WITH RESPECT TO

X - AXIS :: 90.0000011

Y - AXIS :: 180.000002

Z - AXIS :: 90.0000011

CENTER POINTS

(x1, y1, z1) :: 140. 0. 10.

(x2, y2, z2) :: 140. -80. 10.

COEFFICIENTS (A,B,C,D,E,F,G,H,J,K)

6400. 0. 6400. 0. 0.

0. -1792e3. 0. -128000. 12544e4.

COEFFICIENTS OF TOP & BOTTOM PLANE EQUATIONS

(P1, Q1, R1, S1) :: 0. -100. 0. 0.

(P2, Q2, R2, S2) :: 0. -100. 0. 8000.

>>>

CYLINDER NUMBER : 3

CYLINDER RADIUS (r) :: 10. & CYLINDER LENGTH (l) :: 80.

ANGLES WITH RESPECT TO

X - AXIS :: 90.0000011

Y - AXIS :: 180.000002

Z - AXIS :: 90.0000011

CENTER POINTS

(x1, y1, z1) :: 6.12303177E-016 0. 110.

(x2, y2, z2) :: 6.12303177E-016 -80. 110.

COEFFICIENTS (A,B,C,D,E,F,G,H,J,K)

6400. 0. 6400. 0. 0.

0. -7.83748066E-012 0. -1408000. 76800000.

COEFFICIENTS OF TOP & BOTTOM PLANE EQUATIONS

(P1, Q1, R1, S1) :: 0. -100. 0. 0.

(P2, Q2, R2, S2) :: 0. -100. 0. 8000.

>>
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

CYLINDER NUMBER : 4

CYLINDER RADIUS (r) :: 10. & CYLINDER

LENGTH (l) :: 80.=

ANGLES WITH RESPECT TO

X - AXIS :: 90.0000011

Y - AXIS :: 180.000002

Z - AXIS :: 90.0000011

CENTER POINTS

(x1, y1, z1) :: 5.51089037E-015 0. 10.

(x2, y2, z2) :: 5.51089037E-015 -80. 10.

COEFFICIENTS (A,B,C,D,E,F,G,H,J,K)

6400. 0. 6400. 0. 0.

0. -7.05393968E-011 0. -128000. 0.

COEFFICIENTS OF TOP & BOTTOM PLANE EQUATIONS

(P1, Q1, R1, S1) :: 0. -100. 0. 0.

(P2, Q2, R2, S2) :: 0. -100. 0. 8000.

>>

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

CYLINDER NUMBER : 5

CYLINDER RADIUS (r) :: 5. & CYLINDER LENGTH (l) :: 20.

ANGLES WITH RESPECT TO

X - AXIS :: 180.000002

Y - AXIS :: 90.0000011

Z - AXIS :: 90.0000011

CENTER POINTS

(x1, y1, z1) :: 80. -65. 60.

(x2, y2, z2) :: 60. -65. 60.

COEFFICIENTS (A,B,C,D,E,F,G,H,J,K)

----

0. 400. 400. 0. 0.

0. 0. 52000. -48000. 3120000.

COEFFICIENTS OF TOP & BOTTOM PLANE EQUATIONS

(P1, Q1, R1, S1) :: -25. 0. 0. -2000.

(P2, Q2, R2, S2) :: -25. 0. 0. -1500.

>>

ellipsiod.txt

##

THIS IS THE OUTPUT FILE FOR ELLIPSOID SURFACES.

NUMBER OF GEOMETRIES BOUNDED BY ATLEAST ONE ELLIPSOID SURFACE (ELLIPSOID).

OUTPUT FORMAT OF ONE ELLIPSOID:

1. AXIS LENGTHS ALONG X-,Y-,Z- DIRECTIONS WILL BE PROVIDED.

2. CENTER POINT OF ELLIPSOID (x, y, z) WILL BE PROVIDED.

3. ANGLES WITH RESPECT TO X-,Y-,Z- AXIS WILL BE PROVIDED.

4. COEFFICIENTS OF GENERALIZED QUADRATIC EQUATION IN THE FORM OF

AX^2 + BY^2 + CZ^2 + DXY + EYZ + FXZ + GX + HY + JZ + K = 0

A,B,C,D,E,F,G,H,J,K WILL BE PROVIDED.

##

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

ELLIPSOID NUMBER :: 1

AXIS LENGTHS ALONG

X - Axis (A) :: 25.

Y - Axis (B) :: 5.

Z - Axis (C) :: 60.

CENTER POINT

(x, y, z) :: 70. -75. 60.

ANGLES WITH RESPECT TO

X - AXIS :: 90.0000011

Y - AXIS :: 180.000002

Z - AXIS :: 90.0000011

COEFFICIENTS (A,B,C,D,E,F,G,H,J,K)

90000. 2250000. 15625. 0. 0.

0. -12600000. 337500000. -1875000. 1.309725E+010 >>>

Note: Other output files are not provided for restricting the length of the article.

Problem-2}

Input file

$inputs

input = 'INPUT/TEST2.STEP'

option = 'P'

$

C option = 'A' for ALL Surfaces

C option = 'P' for Plane Surfaces

C option = 'S' for Spherical Surfaces

C option = 'C' for Cylindrical Surfaces

C option = 'CO' for Conical Surfaces

C option = 'E' for Ellipsoid Surfaces

C option = 'EC' for Elliptical Cylinder Surfaces

C option = 'T' for Circular Torus

Output files

planes.txt

THIS IS THE OUTPUT FILE FOR PLANAR SURFACES.

NUMBER OF GEOMETRIES BOUNDED BY ATLEAST ONE PLANE(PLANAR SOLID).

OUTPUT FORMAT OF ONE PLANAR SOLID:

1. FOUR CORNER POINTS FOR EACH PLANE.

2. COEFFICIENTS OF PLANE EQUATION IN THE FORM OF :: AX+BY+CZ-D=0

A,B,C,D WILL BE PROVIDED.

##

<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

PLANAR SOLID NUMBER :: 1 & TOTAL NUMBER OF PLANES :: 8

PLANE :: 1

CORNER POINTS

1 : 14.4337567 -25. 0.

2 : 28.8675135 6.9388939E-15 0.

3 : 28.8675135 6.9388939E-15 100.

4 : 14.4337567 -25. 100.

COEFFICIENTS

(A, B, C, D) :: 2500. -1443.37567 0. 72168.7836

PLANE :: 2

CORNER POINTS

1 : 28.8675135 6.9388939E-15 0.

2 : 14.4337567 25. 0.

3 : 14.4337567 25. 100.

4 : 28.8675135 6.9388939E-15 100.

COEFFICIENTS

(A,B,C,D) :: 2500. 1443.37567 0. 72168.7836

PLANE :: 3

CORNER POINTS

1 : 14.4337567 25. 0.

2 : -14.4337567 25. 0.

3 : -14.4337567 25. 100.

4 : 14.4337567 25. 100.

COEFFICIENTS

(A,B,C,D) :: -3.55271368E-13 2886.75135 0. 72168.7836

PLANE :: 4

CORNER POINTS

1 : -14.4337567 25. 0.

2 : -28.8675135 -6.9388939E-15 0.

3 : -28.8675135 -6.9388939E-15 100.

4 : -14.4337567 25. 100.

COEFFICIENTS

(A, B, C, D) :: -2500. 1443.37567 0. 72168.7836

PLANE :: 5

CORNER POINTS

1 : -28.8675135 -6.9388939E-15 0.

2 : -14.4337567 -25. 0.

3 : -14.4337567 -25. 100.

4 : -28.8675135 -6.9388939E-15 100.

COEFFICIENTS

(A, B, C, D) :: -2500. -1443.37567 0. 72168.7836

PLANE :: 6

CORNER POINTS

1 : -14.4337567 -25. 0.

2 : 14.4337567 -25. 0.

3 : 14.4337567 -25. 100.

4 : -14.4337567 -25. 100.

COEFFICIENTS

(A, B, C, D) :: 0. -2886.75135 0. 72168.7836

PLANE :: 7

CORNER POINTS

1 : 14.4337567 -28.8675135 -6.9388939E-15

2 : 28.8675135 -14.4337567 -25.

3 : 14.4337567 25. 100.

4 : -14.4337567 25. 100.

COEFFICIENTS

(A, B, C, D) :: 2790.06351 -1443.37567 777.510585 81937.7646

PLANE :: 8

CORNER POINTS

1 : 14.4337567 -14.4337567 25.

2 : 28.8675135 14.4337567 25.

3 : -14.4337567 -25. 0.

4 : -28.8675135 -6.9388939E-15 0.

COEFFICIENTS

(A, B, C, D) :: -721.687836 360.843918 680.822748 1395.56871

>>